PES UNIVERSITY

100 feet Ring Road, BSK 3rd Stage, Bengaluru 560085

UE17CS303 – Machine Learning Assignment

Report

Low Birth Weight Prediction

PES1201700038	APOORVE GUPTA
PES1201700107	ARAVIND PERICHIAPPAN
PES1201700126	PRATYUSH MISHRA

Problem Statement

Low Birth weight (LBW) acts as an indicator of sickness in new-born babies. LBW is closely associated with infant mortality as well as various health outcomes later in life. Various studies show strong correlation between maternal health during pregnancy and the child's birth weight. Exploit machine learning techniques to gain useful information from health indicators of pregnant women for early detection of potential LBW cases. The forecasting problem should be reformulated as a classification problem between LBW and NOT-LBW classes.

ML Techniques Employed

- **KNN (K - Nearest Neighbours):**

K-nearest neighbours is a non-parametric method used for classification and regression. It is one of the easiest ML techniques used. It is a lazy learning model, with local approximation. It basically does nothing in training phase and gets active in the test phase. The input consists of the k closest training examples in the feature space. The output is a class membership. An object is classified by a plurality vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive integer, typically small). k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The neighbours are taken from a set of objects for which the class is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. We have chosen a k value of 4 and train/test ratio of 0.6 for optimal accuracy with respect to the given data.

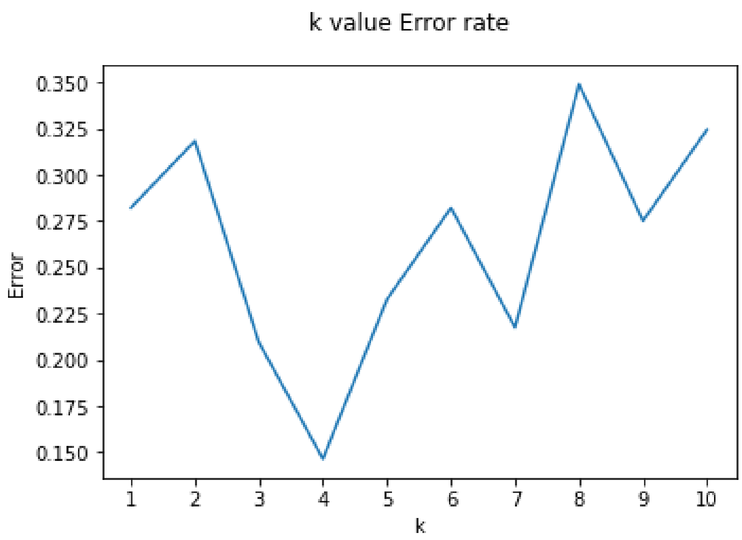
- **Adaboost:**

Adaboost, short for Adaptive Boosting, is a machine learning meta-algorithm proposed by Freund and Schapire. It is an ensemble classifier. Adaboost combines multiple weak classifiers into a single strong classifier. The weak learners in AdaBoost are decision trees with a single split (decision stumps). AdaBoost works by putting more weight on difficult to classify instances and less on those already handled well. Decision stumps are the simplest model we could construct. They guess the same label for every new example, no matter what its features are. AdaBoost is adaptive in the sense that subsequent weak learners are tweaked in favour of those instances misclassified by previous classifiers. The weight-age of each trained classifier at any iteration depends on the accuracy achieved. A classifier with 50% accuracy is given a weight of zero, and a classifier with less than 50% accuracy is given negative weight. Adaboost like random forest classifier gives more accurate results since it depends upon many weak classifiers for final decision. We have chosen a train/test split ratio of 0.8 for optimal accuracy with respect to the given data.

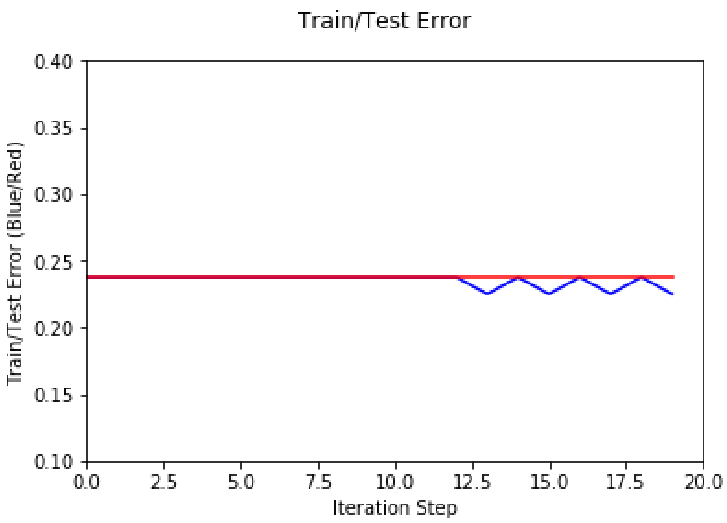
Results and Analysis

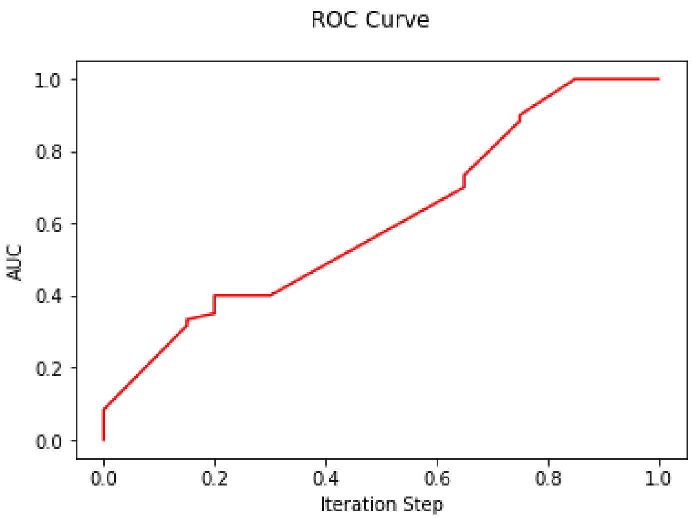
Graphs:

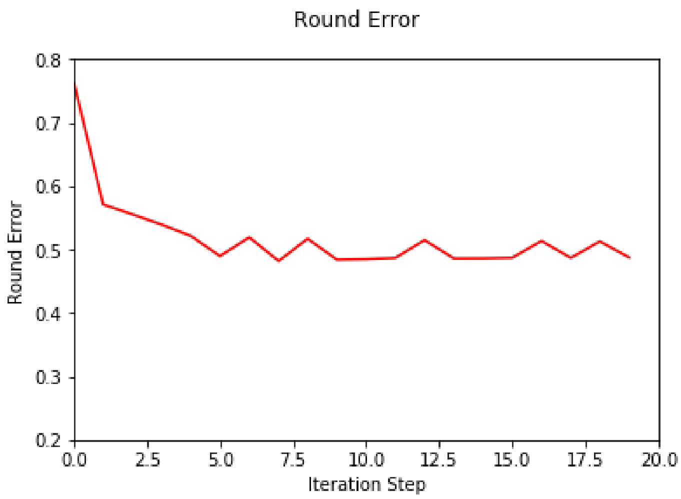
KNN error rate:



Adaboost:


Train/Test Error
Train/Test Error (Blue/Red)
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Iteration Step


ROC Curve
AUC
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Iteration Step


Round Error
Round Error
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Iteration Step

The models were evaluated based on their accuracy on the random testing dataset. The accuracy of predictions on a sample test-train-split is:

```
Accuracy of KNN = 93.33 %
Accuracy of AdaBoost = 76.19 %
```

Conclusions

- Thus, we have examined the data available and classified the people in two classes LBW and NON-LBW.

- For KNN model, it was observed that k=4 produced the least mean error.

- For KNN model, it was observed that split of 0.6 produced optimal accuracy.

- We observe that KNN performs best on this dataset as there are not many rows and there are a few outliers. By using a heuristically accepted value of k, we get good results on the test data.